RECEIVED
CENTRAL FAX CENTER
AUG 2 0 2007

REMARKS

Claims 1-27 are pending. Claims 1-13 and 17-27 including independent claims 1, 17, and 25 were rejected under 35 U.S.C. 103(a) as being unpatentable over Zaidi (2002/0038401 A1) in view of Heinkle (2004/0015739 A1). Claims 25-27 are rejected as being directed to software per se. The Examiner's withdrawal of the 35 U.S.C. 101 rejection of claims 1-24 is gratefully acknowledged. The Examiner also rejected claims 1-27 under 35 U.S.C. 101 and it is unclear whether the Examiner is maintaining this rejection.

The Examiner rejected independent claims 1, 17, and 25 under 35 U.S.C. 103(a) as being unpatentable over Zaidi in view of Heinkel. However, Zaidi and Heinkel, even if appropriately combined, do not teach or suggest generating a plurality of test designs that allow testing of a design automation tool. Zaidi combined with Heinkel is believed to describe only testing of test designs, not a design automation tool. Although the claims are believed allowable in their current form, the claims are being amended to facilitate prosecution. More specifically, claims 1 and 25 have been amended to recite "wherein a probabilistic function is applied to select modules of different types from the library." Claims 17 has been amended to recite "wherein a submodules of different types are randomly selected from the library."

This recitation is supported in Figures 5 and 6 and associated description. More specifically, "At 607, a probabilistic function is applied to select modules of different types from the library of submodules. For example, the user may specify that various memory modules have a 25 percent chance of being selected. A particular phase locked loop module may have a six percent chance of being selected. A probabilistic function is used to select particular submodules from the module library." (page 15, lines 14-19) Furthermore, "According to various embodiments, a user enters the number of designs to generate and a list of hardware families or classes from which a design family will be chosen along with probabilities associated with each family. The user can also enter basic parameters for design generation by family, such as general size or I/O structure. (page 13, line 32 – page 14, line 2) "Randomized and diverse test designs with realistic attributes are automatically generated to allow comprehensive testing of design automation tools such as synthesis, simulation, and place and route tools used to implement designs on electronic devices." (Abstract)

Zaidi and Heinkel do not teach of suggest using any probability function and particularly do not teach or suggest using any probability function to "select modules of different types from the library." In fact, Zaidi and Heinkel teach away from using a probability function. Zaidi and Heinkel provide a deterministic test of a particular test design. It would not make sense for Zaidi and Heinkel to probabilistically select a component for a test system as it would not validate a particular test design under test. By contrast, the techniques of the present invention contemplate testing a design automation tool. In order to sufficiently test a design automation tool, numerous designs probabilistically selected should be run in order to verify the operability of the design automation tool. Zaidi and Heinkel are not believed to teach or suggest any use of a probability function.

The Examiner rejected claims 25-27 as directed to software per se. The Applicants respectfully disagree that claims 25-27 are directed to software per se. Claims 25 recite means elements such as "means for selecting" and "means for applying." To facilitate prosecution, claims elements have been amended to recite "processing means" and "interface means." These means elements provide structural and functional interrelationships between functional elements and the rest of a system such as a computing processing system and is believed to be statutory.

The Examiner argues that Zaidi describes generating multiple test designs in paragraph [0037]. The Applicants respectfully disagree. Zaidi describes in paragraph [0037]. "Most of the block design directories containing RTL source code have a separate subdirectory structure underneath, e.g., "cpubr/", "cpumem/", "dma/", "intctl/", "lcd/", "mc/", "palmbus/", "pio/", "sysctl/", "timer/", and "uart/". The "<block>/sim/" subdirectory includes the simulation tests for the design. The "<block>/synop/" subdirectory includes the Synopsys synthesis scripts and output files for the design. The "<block>/vlog/" subdirectory includes the Verilog RTL source code for the design; if the embodiment language were VHDL, the "<block>/vhdl/" subdirectory includes the VHDL RTL source code for the design."

Paragraph [0037] only describes where the source code for particular components is located. The Zaidi paragraph [0037] does not teach or suggest anything about generating multiple designs or generating test designs.

The Examiner further points to paragraphs [0040]-[0041]. "Each design block has its own separate simulation directory, defined as "<block>/sim/" in the directory structure. Such

directory includes all of the tests exercising that given block in the system. Simulations for the block can be run directly from this directory. Additional tests for the block can be placed into this directory and simulated. New blocks can be easily added into the same environment by adding the same directory structure consisting of the "<newblock>/vlog/" or "<newblock>/vhdl/", and "<newblock>/sim/" directories. Tests exercising this new block in the system would be placed in the "<newblock>/sim/" directory and executed from that directory."

The \sim\ directory includes tests exercising this new block in the system. However, there is still no teaching in Zaidi of generating multiple test designs that allow testing of a design automation tool. Not only does Zaidi not teach or suggest generating the plurality of files in the /sim/ directory, but the files in the /sim/ directory are not a plurality of test designs. The Zaidi /sim/ directory includes multiple files used for testing a block. The Examiner's implication is that the multiple files for testing the block are generated. However, even if these multiple files are generated, these files are not "a plurality of test designs for testing the design automation tool." Zaidi is believed to provide multiple scripts, not test designs. The multiple scripts test only physical blocks such as "any blocks that interact with the main system buses or each other. [0040] Zaidi only possible describes simulations or scripts in the /sim/ directory. There are no generated multiple test designs.

The Examiner also argues that Heinkel teaches generating multiple test designs in paragraphs [0057] - [0060]. However, paragraphs [0057] – [0060] only describes a single "device under test 50." Heinkel in fact does not teach generating a plurality of test designs because Heinkel is configured to test a single DUT (device under test) such as a single ASIC.

By contrast, various embodiments of the present invention allow generation of multiple test designs to allow testing of a design automation tool. In one example, a test design can include a processor, a DSP core, a timer, and a network interface while another generated test design includes a processor, a cryptographic core, and a PIO, and a network interface. Top level modules are instantiated, submodules are parameterized, and interconnection logic is provided for each generated test design. It is respectfully submitted that neither Heinkel nor Zaidi teach or suggest these elements recited in the independent claims.

According to various embodiments, there are test scripts and simulations that are used to test physical blocks. Both Heinkel and Zaidi possibly describe scripts and simulations that are

used to test a physical block or a "new block." By contrast, the independent claims recite generating multiple test designs that allow testing of a design automation tool.

In light of the above remarks, the rejections to the independent claims are believed overcome for at least the reasons noted above. Applicants believe that all pending claims are allowable in their present form. Please feel free to contact the undersigned at the number provided below if there are any questions, concerns, or remaining issues.

Respectfully submitted,
BEYER WEAVER LLP

Godfrey Andrey Kwan
Reg. No. 46,850

P.O. Box 70250
Oakland, CA 94612-0250
(510) 663-1100